IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application No. **10/529,961**

Filed: November 3, 2005

Applicants: Andrew Gordon Williams

Title: **ARRANGEMENT AND METHOD FOR SESSION CONTROL IN WIRELESS COMMUNICATION NETWORK**

Art Unit: 2617

Examiner: Marcos Batista

Attorney Docket: 9010/96542 (02-0073)

Customer No.: 22242

Confirmation No. 7456

This Appeal Brief was electronically filed on February 9, 2010 using EFS-Web.

Mail Stop Appeal Brief -- Patents
Commissioner for Patents
P. O. Box 1450
Alexandria, Virginia 22313-1450

APPEAL BRIEF

Sir:

Pursuant to 37 C.F.R. § 41.37, the Applicants hereby respectfully submit the following Brief in support of their appeal.

TABLE OF CONTENTS

(1) Real Party in Interest

The real party in interest is IPWireless, Inc., a corporation having a primary place of business in San Bruno, California.

(2) Related Appeals and Interferences

There are no related appeals or interferences known to appellant, the appellant's legal representative, or assignee that will directly affect, or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

Claims 1-74 are pending and presently stand at least twice and finally rejected and constitute the subject matter of this appeal.

(4) Status of Amendments

A post-final amendment was submitted on December 9, 2009. The Examiner entered this amendment pursuant to an Advisory Action mailed December 17, 2009.

(5) Summary of Claimed Subject Matter

A concise explanation of this subject matter appears as follows in the form of claim subject matter maps (with corresponding references to the specification by page and line number (or paragraph numbering where appropriate) and to the drawing(s) (if any) by figure number and reference characters where applicable.[1]

Independent Claim 1

	Specification2 Pages/Line Numbers Figure Number/ Reference Character
An apparatus (220) for session control in a wireless communication network, comprising:	FIGS. 4-13

means for detecting (220, 280, 470, 450) requested application-specific packets in a packet stream;	FIGS. 4-13
	Page 10, lines 20-26;
	page 11, lines 29-31;
	page 13, lines 7 and 8;
	page 16, lines 18-25;
	page 18, lines 5-7

means for blocking (220, 470) application-specific packets in the packet stream that are not the requested application-specific packets; and	FIGS. 4-13 Page 18, line 30 – page 19, line 3

[1] It will be understood that this summarization of the claimed subject matter is, in fact, a "summary" and that the Applicants do not represent or intend that this brief presentation, or the accompanying references to the drawings and the specification, comprise an exhaustive presentation in this regard. As always, the claims are to be viewed and interpreted in view of the context of the entire specification sans the Abstract.

2 References to the page and line numbers of the specification refer to the specification as originally filed.

	Specification2 Pages/Line Numbers Figure Number/ Reference Character
means for activating (220, 340, 460), in response to the means for detecting the requested application specific packets, a plurality of packet sessions with application-specific QoS parameters, without requiring explicit cooperation of application software.	FIGS. 4-13 Page 11, line 31- page 12, line 1; page 13, lines 9, 10, and 18-26; page 16, lines 8-16; page 18, lines 18-28

Independent Claim 19

	Specification Pages/Line Numbers Figure Number/ Reference Character
A method for session control in a wireless communication network, comprising:	

detecting requested application-specific packets in a packet stream;	FIGS. 4-13
	Page 10, lines 20-26;
	page 11, lines 29-31;
	page 13, lines 7 and 8;
	page 16, lines 18-25;
	page 18, lines 5-7

blocking application-specific packets in the packet stream that are not the requested application-specific packets; and	FIGS. 4-13 Page 18, line 30 – page 19, line 3

	Specification Pages/Line Numbers Figure Number/ Reference Character
activating, in response to detecting the requested application-specific packets, a plurality of packet sessions with application-specific QoS parameters, without requiring explicit cooperation of application software.	FIGS. 4-13 Page 11, line 31- page 12, line 1; page 13, lines 9, 10, and 18-26; page 16, lines 8-16; page 18, lines 18-28

Independent Claim 40

	Specification Pages/Line Numbers Figure Number/ Reference Character
A computer program element (220) having stored therein program code for session control in a wireless communication network, the program code serving to:	FIGS. 4-13 Page 17, lines 20-30

detect requested application-specific packets in a packet stream;	FIGS. 4-13
	Page 10, lines 20-26;
	page 11, lines 29-31;
	page 13, lines 7 and 8;
	page 16, lines 18-25;
	page 18, lines 5-7

block application-specific packets in the packet stream that are not the requested application-specific packets; and	FIGS. 4-13 Page 18, line 30 – page 19, line 3

	Specification Pages/Line Numbers Figure Number/ Reference Character
activate, in response to detecting the requested application-specific packets, a plurality of packet sessions with application-specific QoS parameters, without requiring explicit cooperation of application software.	FIGS. 4-13 Page 11, line 31- page 12, line 1; page 13, lines 9, 10, and 18-26; page 16, lines 8-16; page 18, lines 18-28

(6) Grounds of Rejection to be Reviewed on Appeal

Claim 40 was rejected under 35 U.S.C. 112, 1st paragraph. Claims 1-10,13-16,18-28,31-34,36-40 and 43-74 are rejected under 35 U.S.C. 103(a) as being unpatentable over Suumäki et al. (US 684761081), hereafter "Suumäki," in view of Jungck et al. (US 20060029104 A1), hereafter "Jungck." Claims 11 and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Suumäki et al. (US 6847610 81), hereafter "Suumäki," in view of Jungck et al. (US 20060029104 A1), hereafter "Jungck," further in view of Dorenbosch et al. (US 20030235184 A1), hereafter "Dorenbosch ." Claims 12, 17, 30 and 35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Suumäki et al. (US 6847610 81), hereafter "Suumäki," in view of Jungck et al. (US 20060029104 A1), hereafter "Jungck," further in view of Fenton et al. (US 20030193967 A1), hereafter "Fenton." Claims 41 and 42 are rejected under 35 U.S.C. 103(a) as being unpatentable over Suumäki et al. (US 6847610 B1), hereafter "Suumäki," in view of Jungck et al. (US 20060029104 A1), hereafter "Jungck," further in view of Boyle et al. (US 20050235349 A1), hereafter "Boyle." The Applicant disputes these rejections.

(7) Argument

Rejections under 35 U.S.C. § 112, first paragraph

Claim 40 was rejected under 35 U.S.C. 112, 1st paragraph. In particular, the Examiner objected to this claim as comprising a single-means claim. Pursuant to an after-final amendment, which amendment has been entered, claim 40 now reads as follows:

> A computer program element having stored therein program code for session control in a wireless communication network, the program code serving to:
>
> detect requested application-specific packets in a packet stream;
>
> block application-specific packets in the packet stream that are not the requested application-specific packets; and

activate, in response to detecting the requested application-specific packets, a plurality of packet sessions with application-specific QoS parameters, without requiring explicit cooperation of application software.

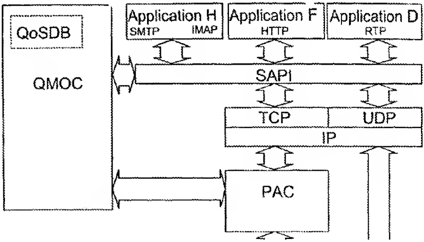
To the extent that this claim might have previously been characterized as a single-means claim, such is no longer the case. At the very least, for example, this claim no longer presents any "means" whatsoever. We therefore respectfully submit that claim 40 is compliant with the requirements of 35 U.S.C. 112, 1st paragraph.

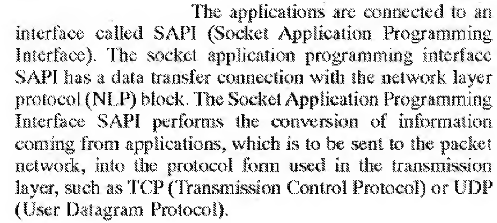
Rejections under 35 U.S.C. § 103(a)

As all of the rejections of the claims under 35 U.S.C. 103(a) are based upon Suumäki, we believe it will be helpful to first briefly discuss Suumäki's teachings prior to considering the merits of these rejections.

The Suumäki reference

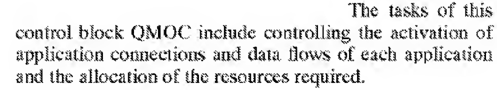
Suumäki discloses a method for optimizing data transmission in a packet switched wireless data transmission system. More particularly, Suumäki presents an approach that is applied with respect to any of a variety of applications that likely have differing quality of service (QoS) requirements or preferences. As shown in Suumäki's FIG. 3A (reproduced in relevant part at the right), these teachings make use of a socket application programming interface (SAPI) that interacts with each of the available applications (such as applications D, F, H as illustrated). More particularly, and as set forth at column 7, lines 19-28 of Suumäki:





> The applications are connected to an interface called SAPI (Socket Application Programming Interface). The socket application programming interface SAPI has a data transfer connection with the network layer protocol (NLP) block. The Socket Application Programming Interface SAPI performs the conversion of information coming from applications, which is to be sent to the packet network, into the protocol form used in the transmission layer, such as TCP (Transmission Control Protocol) or UDP (User Datagram Protocol).

This SAPI, in turn, couples to a so-called quality of service management and optimization control (QMOC) (also shown above in FIG. 3A). As explained at Column 7, lines 38-41 of Suumäki:



> The tasks of this control block QMOC include controlling the activation of application connections and data flows of each application and the allocation of the resources required.

The Examiner notes (and we agree) that Suumäki discloses the idea of providing quality of service parameters to be used with application-specific connections even when the application itself does not provide those QoS parameters. Suumäki discloses, for example, that such parameters might be so-called default parameters residing in a database that is available to the aforementioned QMOC.

What Suumäki does not disclose, however, is an ability to *activate* an application connection without the explicit cooperation of the corresponding application. Merely identifying QoS parameters without the explicit cooperation of the corresponding application, of course, does not, in and of itself, constitute "activating" an application connection. Instead, Suumäki clearly discloses, in each and every one of his general and specific examples, that activation of an application connection requires the explicit cooperation of the corresponding connection with the QMOC.

In particular, Suumäki repeatedly teaches this sequence of events:

(1) An application is started;

(2) The SAPI reacts to this by communicating directly with this application to understand a corresponding communication requirement and forwarding the corresponding information to the QMOC;

(3) The QMOC, one way or the other, provides some QoS parameters to be used with this communication session;

(4) The QMOC uses the foregoing information to activate the corresponding communication context.

Accordingly, it is clear that Suumäki completely relies upon the initial cooperation of these applications and that without such cooperation a corresponding packet session cannot be activated.

Claim 1

Amongst other things, claim 1 specifies activating packet sessions without requiring the explicit cooperation of the application software. This, Suumäki does not disclose. Claim 1 further specifies that such activation is done in response to detecting application specific packets. This, too, Suumäki does not disclose. Instead, Suumäki's applications are specifically required to themselves initiate such activation; it would be hard to imagine a clearer case of "explicit cooperation."

Furthermore, although Suumäki does disclose the idea of gathering QoS parameters from sources other than the application when the application itself is silent in these regards, Suumäki nevertheless teaches that this occurs only in response to a direct initiation of the process by the application. Accordingly, we also submit that Suumäki cannot be fairly read as teaching that QoS parameters are utilized without requiring the explicit cooperation of the application software as Suumäki's QMOC will not undertake the described actions in the absence of this activity being initiated by the application itself.

Claim 1's requirements in these regards are clear:

> [M]eans for activating[3], in response to the means for detecting the requested application specific packets, a plurality of packet sessions with application-specific QoS parameters, without requiring explicit cooperation of application software.

Accordingly, as Suumäki fails to offer any teachings in these regards, we respectfully submit that Suumäki cannot be fairly viewed as anticipating such recitations. The Jungck reference contains nothing to address such shortcomings, nor does the Examiner suggest otherwise. We therefore respectfully observe that no combination of these references, regardless of how obvious or unobvious those combinations might be, will yield the recitations of claim 1.

Independent claims 19 and 40

Claim 19 is a method claim counterpart to claim 1 while claim 40 is a Beauregard-style counterpart to these claims. Accordingly, both of these claims include limitations essentially identical to those described above for claim 1. In particular, claim 19 provides:

> [A]ctivating, in response to detecting the requested application-specific packets, a plurality of packet sessions with application-specific QoS parameters, without requiring explicit cooperation of application software.

and claim 40 provides:

[3] Claim 1's "means for activating" comprises the disclosed stateful packet inspector session monitor.

> [A]ctivate, in response to detecting the requested application-specific packets, a plurality of packet sessions with application-specific QoS parameters, without requiring explicit cooperation of application software.

Therefore, the points raised above with respect to claim 1 are applicable here as well. These points will not be reiterated for the sake of brevity aside from noting that no combination of Suumäki with any of the other references will yield such recitations.

Dependent claims 2-18, 20-39, and 41-74

These dependent claims are all ultimately dependent upon one of independent claims 1, 19, or 40. While the applicant believes that other arguments are available to highlight the allowable subject matter presented in various ones of these dependent claims, the applicant also believes that the comments set forth herein regarding allowability of the independent claims are sufficiently compelling to warrant present exclusion of such additional points for the sake of brevity and expedited consideration.

Conclusion

Claims 1-74 are not rendered obvious by the references of record.

Respectfully submitted,
FITCH, EVEN, TABIN & FLANNERY

Dated: February 9, 2010

Steven G. Parmelee

Registration No. 28,790

120 South LaSalle Street, Suite 1600
Chicago, Illinois 60603-3406
Telephone (312) 577-7000
Facsimile (312) 577-7007

(8) Claims Appendix

1. An apparatus for session control in a wireless communication network, comprising:

means for detecting requested application-specific packets in a packet stream;

means for blocking application-specific packets in the packet stream that are not the requested application-specific packets; and

means for activating, in response to the means for detecting the requested application specific packets, a plurality of packet sessions with application-specific QoS parameters, without requiring explicit cooperation of application software.

2. The apparatus of claim 1 further comprising means for deactivating at least one of the plurality of packet sessions.

3. The apparatus of claim 1 wherein the wireless communication network comprises a UMTS radio access network.

4. The apparatus of claim 1, wherein the packet sessions comprise Packet Data Protocol (PDP) contexts.

5. The apparatus of claim 1 wherein the means for detecting comprises stateful inspection means, and the apparatus further comprises session manager means and packet filter means responsive to the stateful inspection means.

6. The apparatus of claim 1, wherein the means for detecting is arranged to inspect uplink packet flows to detect application-specific packet flows, via application-specific control messages.

7. The apparatus of claim 1, wherein the means for detecting is arranged to inspect downlink packet flows to detect application-specific packet flows, via application-specific control messages.

8. The apparatus of claim 1, wherein the packet sessions comprise conversational class PDP contexts.

9. The apparatus of claim 8, wherein the conversational class PDP contexts are arranged to carry Voice over IP (VOIP) traffic.

10. The arrangement apparatus of claim 8, wherein the conversational class PDP contexts are arranged to carry Video over IP traffic.

11. The apparatus of claim 9 wherein the traffic is based on originated calls controlled by Session Initiation Protocol (SIP).

12. The apparatus of claim 9 wherein the traffic is based on originated calls controlled by H.323 protocol.

13. The apparatus of claim 1, wherein the packet sessions comprise streaming class PDP contexts.

14. The apparatus of claim 13, wherein the streaming class PDP contexts are arranged to carry streaming media traffic controlled by Real Time Streaming Protocol.

15. The apparatus of claim 1, wherein the packet sessions comprise interactive class PDP contexts.

16. The apparatus of claim 1, wherein the packet sessions comprise background class PDP contexts.

17. The apparatus of claim 16, wherein the background class PDP contexts are arranged to carry Post Office Protocol-Version 3 (POP3) traffic.

18. The apparatus of claim 16, wherein the background class PDP contexts are arranged to carry Simple Mail Transfer Protocol (SMTP) traffic.

19. A method for session control in a wireless communication network, comprising:

detecting requested application-specific packets in a packet stream;

blocking application-specific packets in the packet stream that are not the requested application-specific packets; and

activating, in response to detecting the requested application-specific packets, a plurality of packet sessions with application-specific QoS parameters, without requiring explicit cooperation of application software.

20. The method of claim 19 further comprising deactivating at least one of the plurality of packet sessions.

21. The method of claim 19 wherein the wireless communication network comprises a UMTS radio access network.

22. The method of claim 19, wherein the packet sessions comprise Packet Data Protocol (PDP) contexts.

23. The method of claim 19, wherein detecting comprises detecting in a stateful inspector, and the method further comprises providing a session manager and a packet filter responsive to the stateful inspection means.

24. The method of claim 19, wherein detecting comprises inspecting uplink packet flows to detect application-specific packet flows, via application-specific control messages.

25. The method of claim 19, wherein detecting comprises inspecting downlink packet flows to detect application-specific packet flows, via application-specific control messages.

26. The method of claim 19, wherein the packet sessions comprise conversational class PDP contexts.

27. The method of claim 26, wherein the conversational class PDP contexts carry Voice over IP (VOIP) traffic.

28. The method of claim 26, wherein the conversational class PDP contexts carry Video over IP traffic.

29. The method of claim 27 wherein the traffic is based on originated calls controlled by Session Initiation Protocol (SIP).

30. The method of claim 27 wherein the traffic is based on originated calls controlled by H.323 protocol.

31. The method of claim 19, wherein the packet sessions comprise streaming class PDP contexts.

32. The method of claim 31, wherein the streaming class PDP contexts carry streaming media traffic controlled by Real Time Streaming Protocol.

33. The method of claim 19, wherein the packet sessions comprise interactive class PDP contexts.

34. The method of claim 19, wherein the packet sessions comprise background class PDP contexts.

35. The method of claim 34, wherein the background class PDP contexts carry Post Office Protocol-Version 3 (POP3) traffic.

36. The method of claim 34, wherein the background class PDP contexts carry Simple Mail Transfer Protocol (SMTP) traffic.

37. The method of claim 19, wherein the method is performed in User equipment (UE).

38. User equipment (UE) for use in a UTRA system, the user equipment comprising the apparatus of claim 1.

39. An integrated circuit comprising the apparatus of claim 1.

40. A computer program element having stored therein program code for session control in a wireless communication network, the program code serving to:

detect requested application-specific packets in a packet stream;

block application-specific packets in the packet stream that are not the requested application-specific packets; and

activate, in response to detecting the requested application-specific packets, a plurality of packet sessions with application-specific QoS parameters, without requiring explicit cooperation of application software.

41. The apparatus of claim 5, wherein detecting in a stateful inspector comprises inspecting packets, implying a state of an application-specific packet session

via control packets and allowing packets for said session to flow through the firewall if said session originated from inside the firewall or otherwise, blocking said session otherwise.

42. The method of claim 23, wherein detecting in a stateful inspector comprises inspecting packets, implying a state of an application-specific packet session via control packets and allowing packets for said session to flow through the firewall if said session originated from inside the firewall or otherwise, blocking said session otherwise.

43. The apparatus of claim 2, wherein the packet sessions comprise Packet Data Protocol (PDP) contexts.

44. The apparatus of claim 3, wherein the packet sessions comprise Packet Data Protocol (PDP) contexts.

45. The apparatus of claim 2, wherein the means for detecting comprises stateful inspection means, and the apparatus further comprises session manager means and packet filter means responsive to the stateful inspection means.

46. The apparatus of claim 3, wherein the means for detecting comprises stateful inspection means, and the apparatus further comprises session manager means and packet filter means responsive to the stateful inspection means.

47. The apparatus of claim 4, wherein the means for detecting comprises stateful inspection means, and the apparatus further comprises session manager means and packet filter means responsive to the stateful inspection means.

48. The apparatus of claim 5, wherein the means for detecting is arranged to inspect uplink packet flows to detect application-specific packet flows, via application-specific control messages.

49. The apparatus of claim 5, wherein the means for detecting is arranged to inspect downlink packet flows to detect application-specific packet flows, via application-specific control messages.

50. The apparatus of claim 2, wherein the packet sessions comprise conversational class PDP contexts.

51. The apparatus of claim 4, wherein the packet sessions comprise conversational class PDP contexts.

52. The apparatus of claim 2, wherein the packet sessions comprise streaming class PDP contexts.

53. The apparatus of claim 4, wherein the packet sessions comprise streaming class PDP contexts.

54. The apparatus of claim 2, wherein the packet sessions comprise interactive class PDP contexts.

55. The apparatus of claim 4, wherein the packet sessions comprise interactive class PDP contexts.

56. The apparatus of claim 2, wherein the packet sessions comprise background class PDP contexts.

57. The apparatus of claim 4, wherein the packet sessions comprise background class PDP contexts.

58. The method of claim 20, wherein the packet sessions comprise Packet Data Protocol (PDP) contexts.

59. The method of claim 23, wherein detecting comprises inspecting uplink packet flows to detect application-specific packet flows, via application specific control messages.

60. The method of claim 23, wherein detecting comprises inspecting downlink packet flows to detect application-specific packet flows, via application specific control messages.

61. The method of claim 20, wherein the packet sessions comprise conversational class PDP contexts.

62. The method of claim 22, wherein the packet sessions comprise conversational class PDP contexts.

63. The method of claim 20, wherein the packet sessions comprise streaming class PDP contexts.

64. The method of claim 22, wherein the packet sessions comprise streaming class PDP contexts.

65. The method of claim 20, wherein the packet sessions comprise interactive class PDP contexts

66. The method of claim 22, wherein the packet sessions comprise interactive class PDP contexts.

67. The method of claim 20, wherein the packet sessions comprise background class PDP contexts.

68. The method of claim 22, wherein the packet sessions comprise background class PDP contexts.

69. The method of claim 20, wherein the method is performed in User equipment (UE).

70. The method of claim 21, wherein the method is performed in User equipment (UE).

71. The method of claim 22, wherein the method is performed in User equipment (UE).

72. The method of claim 23, wherein the method is performed in User equipment (UE).

73. The method of claim 24, wherein the method is performed in User equipment (UE).

74. The method of claim 25, wherein the method is performed in User equipment (UE).

(9) Evidence Appendix

None

(10) Related Proceedings Appendix

None